UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA

Sistema de Análise e Gestão de Vulnerabilidades: Implementação numa Instituição Bancária

José Miguel Coutinho Marques de Almeida

Mestrado em Informática

Trabalho de Projeto orientado por:
Prof. Doutor Hugo Alexandre Tavares Miranda

2019

Agradecimentos

Gostaria de agradecer ao meu orientador, Prof. Hugo Miranda, por toda a ajuda, aconselhamento e apoio despendidos ao longo deste projeto, e ao meu supervisor Sérgio de Sá por me permitir desenvolver este trabalho na EY.

A toda a equipa que me acompanhou ao longo deste projeto, os restantes estagiários que partilharam as dificuldades e experiências, e os meus pares e seniores, que me ajudaram e encaminharam em tantos momentos, e que pude ter o prazer de chamar amigos, Alessandra, Fernando, Rui, Cláudio, Marisa, Vanessa, Débora, Miguel, Radu, Pedro, Hugo, Manuel, João, Luis e Diogo.

Aos que se mantiveram sempre por perto, ainda que longe fisicamente, Isabela, Ana e Daniela, não é fácil pôr por palavras o que significam e o quanto vos agradeço. Não tentarei, aqui.

Um enorme obrigado aos meus pais, pelo apoio e por tudo o que me permitiram. Nunca conseguirei demonstrar o apreço que tenho por vocês e tudo o que vos devo.

Aos meus pais.

Resumo

Num mundo extremamente interligado e dependente das redes de computadores, as tentativas de intrusão em sistemas informáticos tornam-se mais generalizadas necessitando que as equipas de segurança tenham de estar em constante alerta para qualquer indício de ataque. Este processo torna-se difícil sem um conhecimento abrangente dos possíveis vetores de ataque e dos pontos vulneráveis da organização. Para reduzir o impacto na hora do ataque, é crucial existir preparação que permita descobrir os pontos vulneráveis e corrigir ao máximo os mesmos. No entanto, é extremamente difícil, ou até mesmo impossível fechar todas as lacunas e resolver todas as vulnerabilidades. Por isso, é necessário tomar decisões sobre quais as vulnerabilidades mais relevantes que expõem a organização e as suas operações críticas.

De forma a agilizar este processo estão-se a desenvolver sistemas de análise e gestão de vulnerabilidades capazes de analisar os sistemas e recursos de rede encontrando configurações incorretas e outros tipos de vulnerabilidades, e determinando o seu nível de criticidade, posteriormente gerando relatórios a que as equipas de segurança têm acesso, permitindo priorizar o processo de reparação de vulnerabilidades.

Este projeto consiste no desenho, implementação e avaliação de um sistema de análise e gestão de vulnerabilidades no contexto de uma organização bancária.

Palavras-chave: Cibersegurança, vulnerabilidade, sistema de análise e gestão de vulnerabilidades, análise de vulnerabilidades, criticidade

Abstract

In an extremely interconnected and computer network dependent world, informatic systems intrusion attempts have become more and more generalized requiring security teams to be on constant alert for any signs of attack. This process becomes increasingly more difficult if there is not enough coverage of the possible attack vectors and vulnerable points in the organization. In order to reduce the impact during an attack, it is critical that the preparation exists to allow for the discovery and remediation of vulnerable assets. It is, however, extremely difficult, even impossible, to close every hole and resolve every issue. Because of that, it is necessary to prioritize and decide which vulnerability affects the integrity of the organization and its most critical operations and assets.

With the goal of making this process more agile, today, companies and organizations are developing vulnerability assessment management systems capable of assessing systems and network assets finding security misconfigurations, assigning severity levels, and generating reports that the security teams access, allowing the prioritization in the vulnerability remediation process.

This thesis project will consist in the design, implementation and evaluation of a vulnerability assessment management system in the context of a banking organization.

Keywords: Cybersecurity, vulnerability, vulnerability assessment management system, vulnerability assessment, severity

Conteúdo

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

Uma das funções mais relevantes da cibersegurança é a análise das vulnerabilidades que decorrem dos vários processos de desenvolvimento ou configuração das aplicações e sistemas informáticos. O objetivo é reduzir o impacto das vulnerabilidades, se possível, eliminando-as. Estas vulnerabilidades são, na sua grande maioria, criadas de forma acidental, surgindo após configurações incorretas ou o não cumprimento de boas práticas, muitas vezes documentadas, para desenvolvimento de software e aplicações.

Na avaliação de vulnerabilidades utilizam-se inúmeras ferramentas que permitem evidenciar possíveis configurações incorretas ou código que não cumpre com as boas práticas levando a pontos de risco vulneráveis a ameaças que podem colocar em causa o normal funcionamento dos sistemas. O projeto OWASP – Open Web Application Security Project - publica regularmente as 10 maiores vulnerabilidades encontradas no desenvolvimento web, no seu OWASP Top Ten. Na última versão de 2017 encontramos vulnerabilidades de injeção de SQL, exposição de dados sensíveis, configurações incorretas de segurança, falhas no controlo de acesso, utilização de componentes com vulnerabilidades conhecidas, entre outros[1]. As configurações incorretas de segurança, posicionadas no quinto lugar da lista, são teoricamente simples de corrigir, mas em redes vastas com inúmeros dispositivos a gestão da multitude de configurações existentes bem como da sua correção torna-se extremamente complexa.

Um sistema de análise e gestão de vulnerabilidades reduz este esforço facilitando não só o processo de avaliação de vulnerabilidades, uma vez que a equipa de segurança terá a informação das vulnerabilidades existentes na sua rede, mas também o processo de reparação pois o sistema será capaz de atribuir níveis de criticidade às vulnerabilidades encontradas priorizando a resolução das mais críticas e descrevendo as recomendações de resolução.

É o foco deste projeto o desenho, implementação e avaliação de um sistema de análise e gestão de vulnerabilidades caracterizado pelas características acima referidas.

Ao longo deste projeto, este mesmo sistema de análise e gestão de vulnerabilidades foi concebido, desenvolvido e implementado na forma de uma prova de conceito (POC – *Proof of Concept*), testada com a sua integração em produtos do cliente, com múltiplas variantes arquiteturais.

Para responder aos requisitos da organização cliente, estudaram-se diferentes possibilidades de arquitetura bem como diferentes fornecedores dos produtos para a implementação do sistema.

Como suporte à implementação de um sistema de análise e gestão de vulnerabilidades é necessária a definição de metodologias e normas que balizem a utilização do mesmo, bem como os passos a seguir para a resolução das falhas identificadas. Como tal, desenvolveu-se, em conjunto com a implementação do sistema, uma Norma de Gestão de Vulnerabilidades e Patch Management, utilizada para descrever como analisar os produtos e sistemas existentes na infraestrutura do cliente, e o que fazer para gerir o processo para a sua remediação.

A descrição aprofundada deste trabalho será realizada ao longo deste documento, com especial foco nas tecnologias utilizadas, fornecedores e na arquitetura definida.

1.1 Motivação

Os riscos inerentes a um mundo cada vez mais informatizado tornam a cibersegurança um tema extremamente relevante nos dias de hoje. Com o aumento dos ataques e exploração de vulnerabilidades por parte de indivíduos ou entidades mal-intencionadas, é de extrema importância encontrar possíveis falhas no correto funcionamento de sistemas críticos das empresas, de forma a minimizar o risco de disrupção por parte de eventuais ataques.

Como parte do 2º ano do Mestrado em Informática pela Faculdade de Ciências da Universidade de Lisboa, que consiste na realização de um trabalho autónomo de natureza científica ou profissionalizante, este projeto é desenvolvido em comum com um estágio curricular realizado na área de cibersegurança de uma instituição externa de acolhimento. O projeto compreende o desenvolvimento de um sistema de análise e gestão de vulnerabilidades para um banco cliente da instituição de acolhimento.

1.2 Objetivos

Este projeto tem como objetivo a conceção, definição e desenvolvimento de um sistema de análise e gestão de vulnerabilidades, permitindo facilitar a sua descoberta e

auxiliar na priorização da remoção das mesmas. A implementação deste sistema é integrada numa prova de conceito elaborada numa instituição bancária para uma possível futura extensão a todo o parque do mesmo.

O sistema implementado deverá dar resposta a diversos requisitos funcionais e de segurança acordados e definidos em conjunto com a instituição bancária. Inicialmente era esperado o desenvolvimento de uma prova de conceito e de um sistema final distintos, mas com será explorado na secção de planeamento, ambos foram fundidos numa única fase de desenvolvimento.

No final da dissertação, será feita uma análise do cumprimento de objetivos pelo sistema como parte da sua avaliação.

1.3 Contributos

Este projeto, desenvolvido de forma integrada num cliente da instituição de acolhimento, pretende definir a gestão e análise de vulnerabilidades num sistema automatizado, contribuindo para encontrar os obstáculos mais comuns ao seu desenho, definição arquitetural e desenvolvimento na realidade de uma organização bancária, associado às suas necessidades regulamentares.

1.4 Estrutura do documento

O presente documento encontra-se organizado em cinco capítulos:

Capítulo 1 - O primeiro dedicando-se à introdução, descrevendo o contexto e o tema a desenvolver, a motivação do projeto, objetivos, e a descrição da estrutura do documento.

Capítulo 2 - No segundo capítulo segue-se uma descrição pormenorizada dos objetivos do projeto desenvolvido, o seu contexto, metodologia. É explorada a importância do tema tratado e da sua aplicação com foco nos sistemas bancários, bem como a definição de alguns dos conceitos mais relevantes, é ainda apresentado o planeamento do projeto confrontando-o com os desvios verificados ao logo do mesmo.

Capítulo 3 – O terceiro capítulo debruça-se sobre o trabalho realizado durante a especificação dos requisitos do sistema, o trabalho de conceção da sua arquitetura, definição e desenvolvimento da *demo* (prova de conceito), bem como a elaboração de um processo de gestão de vulnerabilidades e *patch management* para aplicação e adaptação do sistema de gestão de vulnerabilidades à realidade da instituição em que se insere.

Capítulo 4 - No quarto capítulo é feita uma avaliação do sistema desenvolvido e do seu sucesso no cumprimento dos requisitos e objetivos iniciais.

Capítulo 5 – Por fim, é apresentado um sumário do trabalho realizado, conclusões e ainda, uma descrição de potenciais melhorias futuras.

Capítulo 2

Deteção Automática de Vulnerabilidades

2.1 Gestão de Vulnerabilidades

Podemos definir uma vulnerabilidade, de acordo com o NIST[1], como uma fraqueza num sistema de informação, procedimentos de segurança, controlos internos ou na sua implementação que possa ser explorada ou despoletada por uma ameaça [2].

A descoberta e análise de vulnerabilidades, conhecida em inglês como *vulnerability assessment*, pode ser realizada com recurso a inúmeras ferramentas como *Nessus*[3], *Nmap*[4], *Burp*[5], *OpenVas*[6], *Nexpose*[7], *Metasploit*[8], entre outras, que são na sua maioria *open-source* (as que não o são geralmente possuem *Community Editions*[2]), criadas e suportadas pela comunidade, o que tem consequências positivas e negativas. Por um lado, não é difícil licenciar ferramentas para utilizar em aprendizagem ou até em projetos pontuais, em que apenas exista uma fase de análise de vulnerabilidades. Mas no caso de empresas e organizações de maior dimensão, é ingerível basear o processo de *vulnerability assessment* em ferramentas que requerem processos demasiado manuais e que, embora sejam altamente configuráveis, no caso das ferramentas *open-source*, podem não responder às necessidades específicas de cada sistema organizacional. Segue, portanto, a necessidade de criar processos e métodos facilmente escaláveis e configuráveis, que deem resposta às necessidades de cobertura destas organizações. Quando falamos de cobertura, estamos perante a capacidade de abrangência da deteção de vulnerabilidades, quer seja em função da infraestrutura existente na organização, ou mesmo das aplicações e software que esta utiliza.

Na análise de vulnerabilidades estamos geralmente perante duas grandes áreas: análise aplicacional e análise de infraestrutura. A análise aplicacional compreende testes executados a aplicações como forma de determinar se estas têm vulnerabilidades conhecidas e se as suas vulnerabilidades têm formas práticas ou teóricas de serem exploradas para conseguir acesso indevido (não autorizado ou fora do âmbito do acesso permitido) ou outro tipo de atividade maliciosa que potencie uma ameaça à aplicação (*penetration testing)*. Análise de infraestrutura tem objetivos semelhantes aos da análise aplicacional, embora focando a sua ação nas configurações dos recursos físicos da rede.

---

[1] *National Institute of Standards and Technology* do *U.S. Department of Commerce*

[2] Estas versões existem em *software* proprietário, com licenças comerciais (*Enterprise*) para projetos maiores, e, na sua versão *open source* estão geralmente limitadas na sua funcionalidade ou alcance.

A recolha de vulnerabilidades não tem de funcionar em separado dos restantes processos de segurança da organização. Pode, por exemplo, ser utilizada para auxiliar posteriormente nos processos de SIEM[3], focando nas áreas de maior risco e passíveis de serem expostas por ameaças externas.

2.2 *Patch Management*

Como previamente discutido, a gestão de vulnerabilidades implica não só a descoberta de vulnerabilidades, mas também os processos de correção das mesmas. O principal processo nesta fase é geralmente denominado pelo seu termo original em inglês *Patch management* – gestão de atualizações – e compreende os procedimentos necessários à manutenção das infraestruturas e sistemas atualizados, face às constantes evoluções de ataques e explorações de vulnerabilidades encontradas.

Uma vez que se baseia na instalação de atualizações de *software*, o *patch management* é um processo potencialmente disruptivo, que, como tal, necessita de cuidadosa preparação e definição para reduzir o risco subjacente ao mesmo, nomeadamente nas possíveis quebras de serviço associadas à sua aplicação.

Ao longo do projeto de desenvolvimento do sistema de análise e gestão de vulnerabilidades, foram também desenvolvidos uma norma e processo de patch management para suportar as metodologias já existentes na organização, mas feitas de forma ad-hoc, não totalmente isento de falhas, que podem originar disrupção nos serviços e que, por isso, é importante formalizá-los.

Uma vez que as quebras de serviço devem ser reduzidas a um mínimo e a implementação de atualizações acarreta riscos de disrupção, esta acaba por ter uma relevância inferior nas organizações, e o cuidado dado a este procedimento é igualmente inferior. Considerando os riscos subjacentes a uma não aplicação correta de um ciclo de *patch management*, a redução da dificuldade da gestão de vulnerabilidades e *patch management*, e dos riscos de disrupção, são uma mais-valia para qualquer empresa.

A implementação e avaliação desta norma não é abrangida por este documento, sendo um processo contínuo que se mantém ainda após a elaboração desta dissertação.

[3] *Security Information and Event Management* – Gestão de eventos e informação de segurança. Embora o projeto não tenha a interação do sistema de gestão de vulnerabilidades com um SIEM, este é um passo futuro previsto que deverá ter início após a conclusão do projeto de implementação do sistema de gestão de vulnerabilidades.

2.3 Análise e gestão automatizada de vulnerabilidades

A aplicação de processos de análise e gestão de vulnerabilidades envolve um trabalho que incorpora inúmeras ferramentas e procedimentos para a análise de vulnerabilidades e posterior correção das mesmas.

É possível implementar estes processos sobre métodos e controlos totalmente manuais. No entanto, a realidade de uma organização de grande dimensão, particularmente tratando-se de uma instituição bancária, considerando não só a extensão da infraestrutura que suporta toda a sua atividade de negócio, mas ainda as necessidades regulamentares existentes (por exemplo SWIFT[9] e PSD2[10][11]), fomentam a criação de sistemas automatizados que tratem da maioria do esforço necessário.

Existem variadas soluções para suportar a gestão de vulnerabilidades de uma forma automatizada, sendo o principal foco das mesmas a realização de análises infraestruturais periódicas, a qualificação das mesmas de acordo com a sua criticidade e a recolha de informação de tendências de correção de vulnerabilidades. Esta informação assiste as equipas na manutenção corretiva de sistemas que possam estar vulneráveis, ao auxiliar na sua priorização, e facilitando a análise do esforço de correção ao longo do tempo.

Já aqui referimos soluções que assistem as equipas de segurança de informação no seu trabalho de análise de vulnerabilidades, sendo que alguns dos fornecedores destes produtos são também responsáveis pelas principais soluções automatizadas de análise de vulnerabilidades. Entre estes incluem-se, de uma forma não exaustiva, Tenable, Rapid7, Qualys e Tripwire.

Os sistemas de análise de vulnerabilidades permitem, no geral, mapear recursos na rede e planear *scans* periódicos aos mesmos, criando um repositório do estado de "saúde" da infraestrutura das organizações face a vulnerabilidades conhecidas. Estas análises abrangem desde a descoberta de necessidades de atualizações das aplicações, análise de configuração de regras de *firewall*, até algoritmos de cifra, cadeias de certificação e protocolos e portos utilizados.

2.4 Planeamento do projeto

Inicialmente, o planeamento do projeto foi distribuído em 4 fases distintas. Sendo que ao longo do projeto estas foram desdobradas e o âmbito de algumas das mesmas alterado.

As fases inicialmente propostas foram as seguintes:

- Fase de definição de requisitos – compreendendo não só a definição inicial de requisitos para o sistema de análise e gestão de vulnerabilidades, mas também a definição arquitetural do mesmo.
- Fase de análise de produto – consistindo na escolha do principal fornecedor para o sistema a implementar.
- Fase de desenvolvimento do piloto – desenvolvimento de um piloto como prova de conceito do sistema a implementar, demonstrando o funcionamento do sistema bem como a sua capacidade de atingir os objetivos propostos, cumprindo com os requisitos estabelecidos.
- Fase de implementação do sistema – implementação da solução final para o sistema de análise e gestão de vulnerabilidades.

Este planeamento sofreu alterações ao longo do desenvolvimento do projeto, sendo a maior diferença notada ao nível das terceira e quarta fases. À medida que o projeto avançou, e ainda durante a definição arquitetural, começou a ser claro que a prova de conceito a desenvolver poderia ser utilizada como base para o sistema implementado no final. Como tal, o piloto foi criado e implementado com utilização futura em mente, tendo a mesma arquitetura e infraestrutura do sistema que se encontra em funcionamento atualmente.

Um dos trabalhos realizados que não foi inicialmente previsto tratou-se da criação da versão inicial da norma e processo de gestão de vulnerabilidades e *patch management*. Este documento, como referido, surgiu com a necessidade de suportar as metodologias já existentes na organização, não determinadas de uma maneira formal, causando resultados mistos na aplicação de *patches*. Este trabalho não constitui necessariamente uma nova fase, sendo que foi feito em simultâneo com várias fases do projeto.

As restantes fases planeadas ocorreram dentro do esperado, com a descrição mais pormenorizada das mesmas em seguida.

A fase de definição de requisitos englobou a definição dos objetivos do projeto, que integrou o desenvolvimento do sistema de análise e gestão de vulnerabilidades, bem como os requisitos funcionais associados a este e a arquitetura final que o sistema deveria integrar.

A fase de análise de produto, que decorreu imediatamente seguindo a definição de requisitos, consistiu na escolha do principal fornecedor para o sistema bem como das potencias tecnologias a adotar para a implementação do mesmo.

Por último, as fases de desenvolvimento do piloto e implementação do sistema, que se simplificaram numa única fase, foram as mais extensas, culminando na conclusão do projeto. Esta fase final teve o propósito de demonstrar o funcionamento do sistema com casos reais, avaliando o cumprimento dos objetivos e requisitos predispostos. Como a implementação foi feita na infraestrutura final, após a utilização do sistema com o propósito de prova de conceito, o mesmo foi utilizado em projetos adicionais que estavam a ser desenvolvidos internamente, para efeitos de análise à infraestrutura.

Capítulo 3

Sistema de Análise e Gestão de Vulnerabilidades

3.1 Definição arquitetural

Para auxiliar o desenvolvimento de um sistema para a análise e gestão de vulnerabilidades, o conhecimento do contexto dos recursos existentes bem como a arquitetura do sistema alvo é muito relevante. No caso particular em que se desenvolve o presente projeto, um cliente da área bancária e financeira, existem preocupações específicas quanto à forma como os dados podem ser analisados e tratados.

Durante a fase inicial do projeto, procurou-se fazer um levantamento de requisitos tendo em vista a criação de uma descrição geral do sistema de gestão de vulnerabilidades a desenvolver, permitindo uma visão global que possibilite a escolha entre possíveis arquiteturas, soluções e fornecedores. Esta fase de definição arquitetural e de requisitos foi desenvolvida ao longo dos dois primeiros meses do projeto, outubro e novembro.

O levantamento de requisitos determinou as seguintes necessidades:

O sistema deve ser capaz de acomodar recursos físicos e virtualizados, localizados *on-premises* e na *cloud*. Esta necessidade surge das características da infraestrutura da organização do cliente. Ainda que a grande maioria dos seus recursos de rede estejam localizados nas suas instalações, existe uma cada vez maior presença de componentes *cloud* utilizados no desenvolvimento aplicacional e no suporte de processos de continuidade de negócio. A análise de recursos virtuais e físicos acarreta especificidade ao nível de licenciamento de ferramentas, uma vez que vários recursos virtuais podem partilhar a mesma estrutura física e serem contabilizados como infraestrutura separada.

Deve possibilitar a deteção de vulnerabilidades em redes protegidas, segregadas, de difícil acesso, em ambientes de desenvolvimento, qualidade, produção e continuidade de negócio, e seria uma mais-valia ter a capacidade de determinar a existência de vulnerabilidades em recursos que se ligam à rede de forma intermitente.

Terá de ser possível realizar análise com credenciais de administração das máquinas, sem comprometer a garantia de segurança e confidencialidade destas credenciais. A esta necessidade alia-se a de garantir diferentes níveis de acesso à informação disponibilizada pelo sistema e a capacidade de garantir separação de funções. Este requisito deriva da necessidade de permitir que as equipas de gestão de risco, segurança, monitorização, desenvolvimento, gestão, entre outras, possam ter acessos personalizáveis a diferentes

tipos de informação. A título de exemplo é fácil compreender que embora a equipa de desenvolvimento de um projeto tenha acesso ao repositório das suas vulnerabilidades, não conseguirá aceder à informação de outro projeto. Ou ainda que a equipa de gestão de segurança tenha acesso a todos os repositórios de vulnerabilidades, mas não conheça as credenciais de administração utilizadas pelas equipas de gestão de sistemas.

Não é necessário que o sistema seja inteiramente *on-prem* podendo ter uma componente *cloud*, desde que os dados sejam guardados em infraestruturas presentes na União Europeia.

Deve ser escalável, permitindo acomodar alterações na infraestrutura da organização, seja o seu expectável crescimento, ou mudanças ao nível arquitetural.

Estes requisitos não derivam somente das características da rede e recursos a analisar, como da própria natureza do cliente para o qual o sistema de gestão se está a desenvolver. Os sistemas de informação bancários estão sujeitos a requisitos regulamentares definidos por organizações nacionais e internacionais como o Banco de Portugal, SIBS, diretivas e regulamentos europeus (por exemplo, PSD2) e SWIFT. Este último, a título de exemplo, requer uma infraestrutura segregada com uma solução de gestão de vulnerabilidade integrada, uma das mais valias da implementação do sistema descrito no presente documento[9].

Ao longo do mês de novembro iniciou-se a fase de análise de produtos, descrita de forma aprofundada ao longo das seguintes secções, que assistiu o processo de escolha de fornecedores para o sistema de análise e gestão. Analisando o mercado, foi possível encontrar inúmeras soluções que se encaixam nas arquiteturas que observaremos nas figuras 1 e 2, sendo que os fornecedores dispõem de soluções direcionadas a clientes que pretendem manter toda a informação *on-premises*, e novas soluções, baseadas em *SaaS*[4], hospedadas na *cloud*, com alguns elementos *on-prem*.

Com base nos requisitos recolhidos determinou-se uma arquitetura geral em que o sistema de gestão de vulnerabilidades se deverá inserir, quer seja totalmente *on-prem* ou que tenha uma componente *cloud*, como podemos verificar nas figuras 1 e 2 respetivamente.

[4] SaaS ou Software as a Service descreve um modelo de disponibilização e licenciamento de *software* em que o produto se encontra hospedado à responsabilidade do fornecedor que o disponibiliza ao cliente *on-demand*. No caso dos fornecedores de ferramentas de análise de vulnerabilidades, este modelo é um misto entre um painel de gestão disponibilizado ao cliente na *cloud* e instrumentos de avaliação presentes nas instalações do cliente.

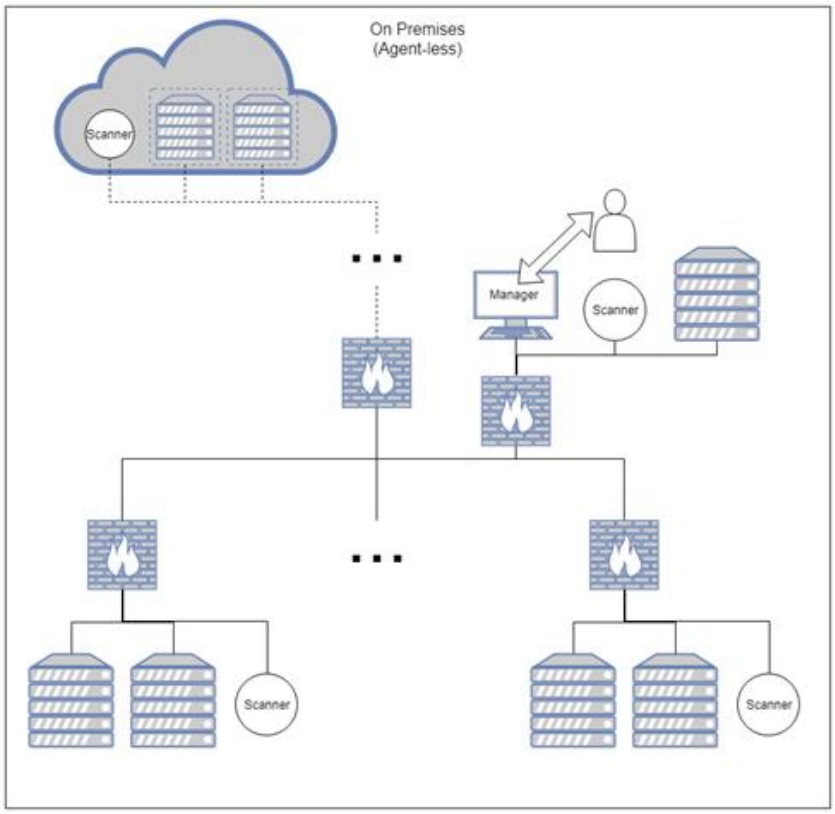

Figura 1 – Arquitetura geral do sistema de gestão sem o recurso a agentes

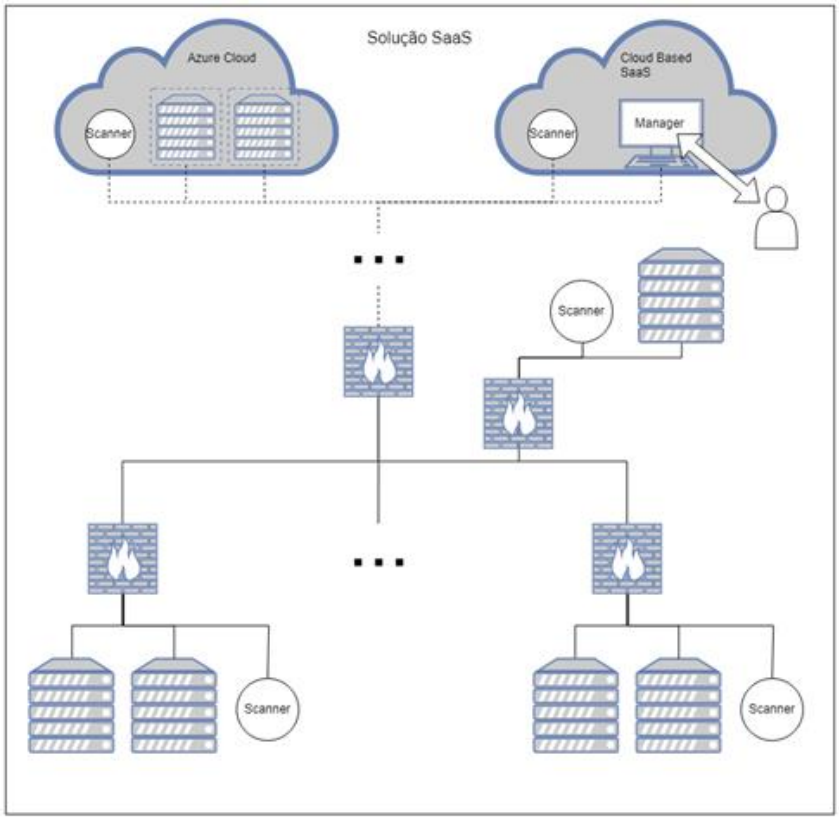

Figura 2 – Arquitetura alternativa para o sistema de gestão, recorrendo a uma solução SaaS

Para analisar recursos que se localizam em redes de difícil acesso ou recursos que se conectam de forma intermitente à rede, os fornecedores propõem soluções que utilizam agentes instalados diretamente nas máquinas a analisar.

A utilização de agentes apresenta algumas vantagens claras face aos métodos de avaliação tradicionais, *agent-less*: Permitem maior visibilidade das vulnerabilidades de cada dispositivo na rede, uma vez que o agente se encontra instalado na máquina com as credenciais necessárias a poder avaliar as configurações do dispositivo em questão. Associada a esta característica verifica-se que há uma maior facilidade na gestão de dispositivos com IPs dinâmicos, ou imagens de *cloud*, uma vez que ainda que o IP do dispositivo possa ser alterado, o agente não é afetado. Isto significa também que as credenciais de acesso a cada dispositivo não têm de ser concentradas num único ponto (tradicionalmente a consola de gestão necessitaria de ter as credenciais suficientes para efetuar uma avaliação correta dos recursos). Por outro lado, ao estarem instalados localmente, tornam mais eficiente a utilização da rede especialmente face a análises não credenciadas, reduzindo o tráfego na mesma ao só necessitarem de enviar os resultados das suas análises.

A utilização de agentes tem, no entanto, algumas desvantagens, não sendo capazes de detetar vulnerabilidades que não se limitam ao próprio dispositivo, como por exemplo vulnerabilidades relacionadas com utilização de certificados SSL, e têm um custo acrescido no licenciamento.

A utilização de agentes ou de um sistema *agent-less* não provoca grandes alterações na arquitetura geral do sistema de gestão de vulnerabilidades. Na figura 3 vemos a implementação de um sistema com agentes na versão *on-prem* face à versão sem agentes. Chamo à atenção para os agentes que surgem representados nas imagens à esquerda A diferença arquitetural é mínima, implicando apenas a instalação dos mesmos nos dispositivos. A solução *SaaS* ilustrada na figura 4 também apenas difere nos agentes presentes nas máquinas que forem necessárias.

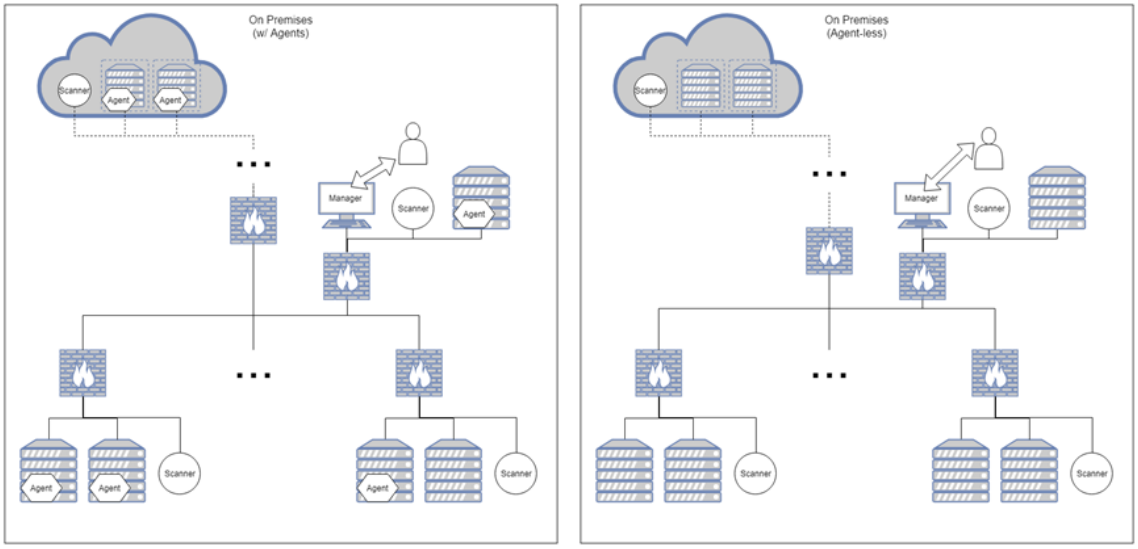

Figura 3 Arquitetura on-prem com agentes (esq.) comparada com uma solução sem agentes (dir.)

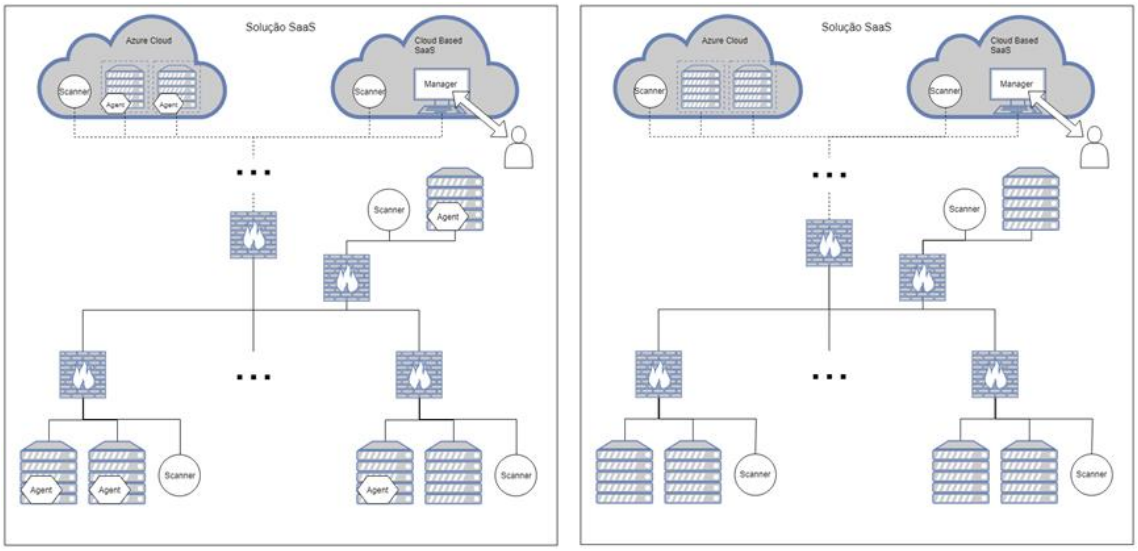

Figura 4 Arquitetura baseada em SaaS com recurso a agentes (esq.) comparada com uma solução sem agentes (dir.)

Uma das preocupações partilhada pelas empresas é a gestão de dados internos em ambientes externos às mesmas. Esta questão leva a que haja uma preferência natural pelo armazenamento dos dados em locais controlados pelas empresas. Como tal, foi decidido que a arquitetura mais confortável para uma instituição bancária se trata de uma solução de infraestrutura *on-premises* em que os dados das análises de vulnerabilidades não são enviados para fora da infraestrutura local, sendo recolhidos num servidor mantido localmente. Desta forma, as soluções SaaS foram descartadas, uma vez que obrigariam a que toda a informação tivesse de ser extraída da rede local.

Considerando ainda a possibilidade de utilização de uma solução com agentes, é relevante ter em conta o contexto da implementação deste tipo de solução na realidade da instituição. A gestão de cada projeto leva a que o esforço inicial de instalação de agentes

e validação de compatibilidades dos mesmos em todas as aplicações seja vasto, o que motivou a que, inicialmente, o sistema seja apenas baseado em sondas ou *scanners* montados na rede, sem o recurso a agentes presentes nas máquinas. Esta decisão leva a alguns desafios ao nível de *scans* efetuados com recurso a credenciais que serão explorados no próximo capítulo.

Segue, portanto, que a arquitetura apropriada para o sistema final seja semelhante ao demonstrado na figura 1, isto é, uma arquitetura sem o recurso a agentes que utiliza scanners posicionados estrategicamente em locais de difícil acesso, que transmitem as suas análises para o coletor central montado na infraestrutura das instalações da empresa. Assim, as principais preocupações de envio de informação para o exterior ficam colmatadas e reduz-se a necessidade de disrupção de cada serviço para aplicação de agentes em infraestrutura de produção.

3.2 Requisitos funcionais da prova de conceito

Como referido, existem necessidades de acordo com a realidade da infraestrutura da organização que implicam a criação de um sistema de análise e gestão de vulnerabilidades que suporte tanto ambientes *on-premises* como na *cloud*.

Para além da estipulação de requisitos gerais de forma a balizar a tipologia arquitetural que o sistema de gestão tem de respeitar, foi ainda necessário escolher objetivos e requisitos para a prova de conceito. Trata-se não somente uma necessidade extraída do facto do sistema ser avaliado para efeitos académicos, mas principalmente pelo facto de se tratar de um investimento considerável, o qual deve ser justificável para a instituição que o pretende implementar.

Inicialmente, tornou-se clara a necessidade de suportar controlo de acessos no sistema, nomeadamente na plataforma utilizada para lançar *scans* e aceder ao repertório de vulnerabilidades identificadas nos diferentes projetos. Não seria somente necessário proteger o acesso através de utilizadores com privilégios para o fazer, mas também potencialmente separar o tipo de acesso por projeto e, ou, responsabilidades. Isto significa que o sistema deve permitir a criação de utilizadores com diferentes níveis de acesso a diferentes recursos num estilo de *Role Based Access Control*, ou RBAC.

Adicionalmente, a natureza do tipo da infraestrutura do cliente, gerida por diversos fornecedores em regime de *managed services*, implica uma cuidadosa gestão de credenciais e contas privilegiadas nos servidores e outros dispositivos da infraestrutura, uma vez que em muitos casos, não sendo a gestão feita internamente, as contas com privilégios de administração são geridas pelos fornecedores. Sendo assim, e considerando

a decisão da não utilização de agentes numa fase inicial, é necessário que a plataforma permita gerir credenciais de uma forma centralizada sem aumentar o risco de exposição das mesmas, ou que, ao fazê-lo, o mesmo seja negligenciável.

A plataforma deve possibilitar o acesso centralizado aos *dashboards* com a informação de vulnerabilidades e deve permitir categorizar esta informação, seja por tipo de *asset* (sistema operativo, função, rede) mas também por outras categorias como projeto, responsável aplicacional, ou outra categorização personalizada que possa ser necessária no futuro. O próprio sistema deve ter a capacidade de enumerar os recursos existentes na rede (*asset discovery*) e categorizá-los corretamente quanto ás suas características conhecidas.

Como resultado das análises efetuadas pelo sistema, sejam estas periódicas e automáticas ou despoletadas manualmente, o sistema deve seguir uma lógica de priorização da correção de vulnerabilidades, atribuindo a estas valores compreensíveis de criticidade. Seria uma mais valia apresentar métodos corretivos para cada uma das vulnerabilidades detetadas.

Associado a este último ponto, a plataforma tem de apoiar diretamente os esforços de *patch management* da organização. Isto requer que as validações de aplicação periódica de *patches* sejam feitas com recurso ao sistema, e que este mantenha um registo e histórico preciso das necessidades de atualização dos sistemas existentes. Esta necessidade foi bastante visível durante o desenvolvimento do projeto, uma vez que a equipa de testes de segurança, com a competência de validar a aplicação de *patches* em âmbito de projeto, tinha de manter um registo extenso e muitas vezes de difícil consulta para controlar as necessidades de cada servidor e assegurar que as mesmas eram cumpridas.

Por vezes, seja por incompatibilidade entre aplicações, falta de certificação de versões de *software* para *appliances*, ou outro tipo de obstáculos à implementação de *patches* e ou correção de vulnerabilidades, é necessário avaliar o risco de não implementação dos mesmos face ao risco operacional de inadvertidamente criar disrupções no normal funcionamento das aplicações organizacionais. Assim, a plataforma deve permitir a análise e gestão de aceitação do risco para os casos que façam sentido.

Finalmente, o sistema de análise e gestão de vulnerabilidades implementado como prova de conceito será utilizado para avaliar os ambientes de qualidade de duas aplicações localizadas em infraestruturas distintas, uma *on-prem* e outra na *cloud*, permitindo determinar os moldes de uma utilização eficaz e eficiente da plataforma para futura implementação.

3.3 Definição de fornecedores

Logo após a conclusão da fase de requisitos, iniciou-se uma fase de investigação para encontrar soluções e produtos existentes que respondessem ao tipo de requisitos previamente discutidos.

De forma a poder comparar as diversas soluções existentes, os requisitos foram projetados em características correspondentes.

Foram estudadas as ofertas de diversos fornecedores verificando-se a semelhança existente no mercado nas diversas soluções de *vulnerability assessment*. Com base no relatório da Gartner *"Gartner Market Guide for Vulnerability Assessment"* os principais fornecedores reconhecidos de soluções de "*vulnerability assessment"* em 2018 foram a Rapid7 e a Tenable.

É também relevante salientar que num mercado em constante evolução como o da análise de vulnerabilidades, é possível que num período de anos, ou até mesmo meses, as soluções sejam alteradas e permitam configurações não inicialmente previstas neste projeto.

A fim de melhor conhecer os produtos de ambos os fornecedores, foram realizadas reuniões de apresentação dos mesmos. Ambos os fornecedores apresentaram as soluções referidas anteriormente e que são direcionadas a clientes que pretendem manter toda a informação *on-premises* (nexpose no caso da Rapid7 e Security Center[5] no caso da Tenable) e novas soluções, baseadas em *Software as a Service*, hospedadas na *cloud*, com elementos de scan ou agentes *on-prem* (insightVM da Rapid7 e Tenable.io da Tenable). Ambos os produtos suportam ambientes Azure, a solução *cloud* utilizada pela organização.

Tendo em consideração a arquitetura definida para o sistema final, uma solução *SaaS* não seria apropriada pelo que se descartaram inicialmente as opções de insightVM e Tenable.io[6], mantendo o foco nas nexpose e Security Center.

A tabela 1 resume as principais características de análise de vulnerabilidades existentes nas soluções *on-prem* da Security Center e nexpose da Tenable e Rapid7 respectivamente, considerando os requisitos da prova de conceito. Foram também tidas em conta características não estritamente necessárias para o cumprimento da prova de conceito, mas que constituem uma mais-valia a possíveis iterações futuras do sistema,

[5] Durante a elaboração do projeto a Tenable renomeou o produto Security Center para Tenable.sc, mantendo todas as funcionalidades originais.

[6] Durante a implementação da prova de conceito, a Tenable apresentou uma nova solução Tenable.io On-prem que pretende ser um misto de uma solução totalmente *SaaS* com uma forte componente disponibilizada na infraestrutura do cliente. Esta solução foi também descartada uma vez que conta com requisitos arquiteturais impossíveis de cumprir numa primeira fase do projeto.

seja o suporte para agentes, dispositivos móveis, *application scanning*, PCI Compliance e integração com sistemas de *tickets*.

	Tenable Security Center	Rapid7 nexpose
RBAC	✓	✓
Armazenamento cifrado de credenciais	✓	✓
Dashboards centralizados	✓	✓
Asset Discovery	✓	✓
Deteção de Malware	✓	✓
Priorização de Remediação	✓	✓
Patch management	✓	✓
Suporte para dispositivos móveis	✓	✓
Suporte para assets em cloud	✓	✓
Suporte para agentes	✓	✗
Suporte para assets virtualizados	✓	✓
"Asset Profiling"(IP, Sistema Operativo, Portos...)	✓	✓
Análise de Risco	✓	✓
Application Security Testing (Pen test, OWASP Top 10, CWE 25)	✓	✓
Suporte para standards COBIT, PCI, HIPAA	✓	✓
Integração com sistemas de tickets	✓	apenas API
Criação de projetos de remediação e "rollover scan"	✓	✗

Tabela 1 Características das soluções consideradas face aos requisitos estabelecidos e características distintivas

A aplicação Tenable Security Center faz uso da plataforma Nessus para o bruto do trabalho que executa. Esta plataforma existe há 25 anos e é uma das mais utilizadas no âmbito de análise de infraestrutura, facilitando a aprendizagem e utilização do sistema. A Rapid7 também faz uso da *framework* Metasploit, amplamente utilizada para exploração de vulnerabilidades que vem pré-instalada na distribuição Kali Linux orientada a *ethical hacking* e *penetration testing*.

As duas soluções apresentam soluções RBAC para controlo de acessos e permitem guardar de forma segura credenciais sem que os utilizadores da plataforma tenham visibilidade sobre as mesmas.

Embora ambas as soluções cumpram com os principais requisitos estudados, a Rapid7 apresenta um foco maior na sua solução *SaaS*, reduzindo as capacidades do produto *on-prem*. A versão nexpose não apresenta sistemas de *tickets* nem permite a criação de projetos de remediação. Adicionalmente não suporta a instalação de agentes sem a extensão do licenciamento de insightVM, algo que, embora não faça parte da primeira iteração do sistema, será muito provavelmente alvo de expansão no futuro. Por outro lado, a experiência prévia das equipas de gestão de segurança da organização com a aplicação de análise de vulnerabilidades Nessus, cria uma natural afinidade e à-vontade para com a plataforma e o tipo de relatórios que a mesma produz.

Existem também diferenças ao nível de licenciamento. A Tenable licencia a sua plataforma, independentemente da utilização de agentes e ou *SaaS*, por *asset*. Isto significa que a utilização de *hosts* virtualizados e contentores *cloud* não serão duplamente cobrados apenas por conterem diferentes IPs. Este esquema de licenciamento é mais favorável do que o proposto pela Rapid7, que apenas suporta este tipo de "*asset-based licensing"* com recurso a agentes instalados.

Tendo em consideração as características descritas, foi determinado seguir o desenvolvimento do sistema de análise e gestão de vulnerabilidades baseado na plataforma Tenable Security Center.

Na figura 5 vemos um diagrama geral do modo de funcionamento do sistema com a adição das sondas Nessus, tanto *on-prem* (dividido entre uma região de *disaster recovery* – DR e uma rede de produção – PRD[7]) como na *cloud* Azure, e da plataforma centralizada Tenable Security Center. Inicialmente o sistema apenas analisará ambientes de qualidade, no entanto, um sistema final necessitará de ter uma visão holística sobre todos os ambientes.

[7] Ao longo deste documento são utilizadas referências a diferentes tipos de ambientes utilizados ao longo do *deployment* de *software* ou *hardware*. Embora possam existir outros, os principais ambientes referidos são desenvolvimento (sigla DEV), qualidade (sigla QUA), produção (sigla PRD) e continuidade de negócio ou *disaster recovery* (siglas DR ou PCN – Plano de Continuidade de Negócio).

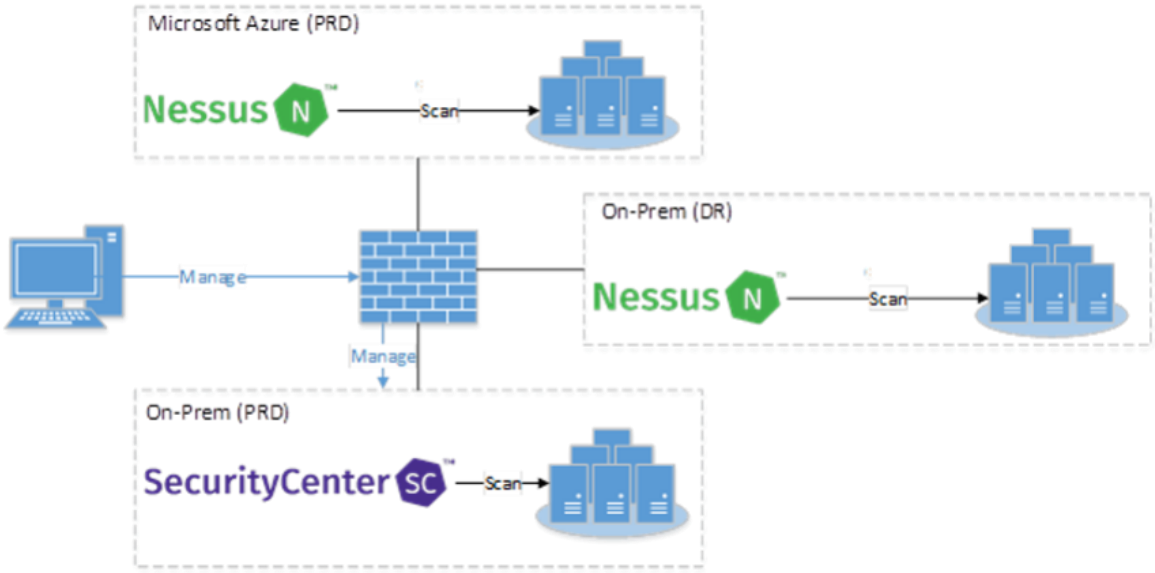

Figura 5 Desenho simplificado da utilização de sondas Nessus em redes segregadas

A imagem mostra, de uma forma simplificada, como redes isoladas podem ter sondas Nessus montadas, utilizadas para aceder aos recursos nessa rede, e transmitirem a informação necessária de volta à plataforma de gestão Security Center através de protocolos seguros e em portos conhecidos. A descrição pormenorizada do funcionamento do sistema será explorada noutro tópico.

3.4 Norma de gestão de vulnerabilidades e *patch management*

Durante o desenvolvimento do presente projeto, iniciou-se a elaboração de uma norma e processo de gestão de vulnerabilidades e *patch management*. Este documento, embora não inicialmente previsto no âmbito do projeto, e não estando concluído aquando do término da prova de conceito, não deixa de ter relevância para o dia-a-dia da utilização do sistema de análise e gestão de vulnerabilidades, pelo que será aqui descrito de uma forma superficial.

Com a implementação do sistema, a análise de vulnerabilidades torna-se periódica, deixando de estar associada apenas às fases finais de implementação de projetos. Assim, é possível ter uma visão contínua sobre a infraestrutura, sobre o repositório de vulnerabilidades da organização e sobre os esforços de remediação existentes. Como tal, é necessário formalizar os processos recorrentes da gestão de vulnerabilidades, abandonando a forma menos abrangente e deliberada de como esta mesma gestão ocorria. Adicionalmente, a visão sobre as vulnerabilidades existentes torna-se global. É possível acompanhar as necessidades da infraestrutura desde a sua conceção, em sede de projeto, e após a sua implementação em produção. Numa organização onde a maioria dos recursos

e investimento se focam no desenvolvimento de projetos, é necessário garantir que a manutenção desses mesmos projetos, particularmente em casos de correção de vulnerabilidades, deve estar prevista. Segue, portanto, a necessidade de uma norma com vista a esta garantia.

É na gestão de *patches* que vemos o maior impacto, naturalmente associado à correção de vulnerabilidades existentes na infraestrutura. Trata-se da fase mais crítica e, geralmente, penosa, envolvendo o maior risco de disrupção do serviço com a instalação de atualizações e a necessidade de reiniciar componentes críticos. Esta gestão já é praticada na organização num regime mensal, mas não existiam métodos de validação e verificação da mesma. Intervenções extraordinárias motivadas por vulnerabilidades severas e componentes críticos sofriam de atrasos significativos na aplicação de *patches* e validação dos mesmos.

Graças à natureza de um sistema de gestão de vulnerabilidades não existe o risco de tornar este processo de correção oneroso, uma vez que existe uma carga associada à priorização de trabalho que é automaticamente feita pelo sistema. O processo deve, portanto, prever a utilização das escalas de esforço e criticidade já calculadas pela plataforma, e reduzir o impacto de intervenções extraordinárias à infraestrutura.

3.5 Implementação da prova de conceito

De seguida, será descrita a implementação do sistema de análise e gestão de vulnerabilidades na sua qualidade de prova de conceito. Como discutido, esta prova de conceito é equiparável à versão final do sistema, diferenciando-se apenas no número de scanners em locais distintos da rede. É, portanto, também objetivo deste sistema assegurar a escalabilidade do mesmo para expansões futuras.

3.5.1 Descrição Geral

O início da implementação do sistema deu-se após a escolha de fornecedor, com o aprovisionamento da infraestrutura necessária. Com a prova de conceito, as máquinas necessárias à montagem do sistema dividem-se entre a infraestrutura local *on-premises* e na *cloud* da Microsoft Azure. Estas máquinas servirão para suportar o sistema final o que significa que deverão ser suficientes para que a infraestrutura seja expandida no futuro.

O principal elo que terá de acomodar esta expansão será o Security Center. Esta máquina terá ligação a todos os *scanners* independentemente da sua localização permitindo um acesso e comando centralizado de todos os scanners. É assim garantida uma visão holística completa da infraestrutura. Esta ligação é feita através de um porto conhecido, 8834 por omissão, e a comunicação entre *scanner* e Security Center é

concretizada utilizando TLS v1.2 para que a mesma seja segura[12]. Assim, apenas é necessário abrir um único porto nas regras das *firewalls* que se encontram entre os scanners e a máquina onde o Security Center reside. No total, a máquina apenas necessita de comunicar com o servidor de e-mail para o envio de alertas por SMTP ou a sua variante com segurança na camada de transporte SMTPS, com o servidor remoto da Tenable para actualização dos *plugins*, e com as máquinas que lhe acederão para acesso à plataforma de gestão[13].

A máquina tem também de acomodar a informação da infraestrutura analisada durante um longo período, por norma, até um ano. A recomendação da Tenable é que para uma gestão ativa de 2 500 *hosts* o Security Center disponha de 4 núcleos de 2GHz de processamento, 8 GB de memória RAM e espaço de armazenamento suficiente para guardar a informação de tendência de vulnerabilidades (90 dias – 125 GB, 180 dias – 250GB). Estes valores, especialmente os de armazenamento, são bastante variáveis uma vez que dependem do número de *hosts* mas também do número de vulnerabilidades encontradas em cada um[14].

O produto não suporta nativamente Network Attached Storage, permitindo instalações em Storage Area Network com latência inferiores a 10 milissegundos, mas aconselhando a utilização de Direct Attached Storage, sendo esta última a solução utilizada[14].

3.5.2 Aprovisionamento e montagem

A instalação do sistema de análise e gestão de vulnerabilidades deu-se com o aprovisionamento da infraestrutura necessária à sua montagem como prova de conceito. Por forma a respeitar a arquitetura identificada na Figura 5 foram estipuladas as criações de *scanners* em Azure e *on-prem* bem como a montagem do portal Security Center numa terceira máquina *on-prem*.

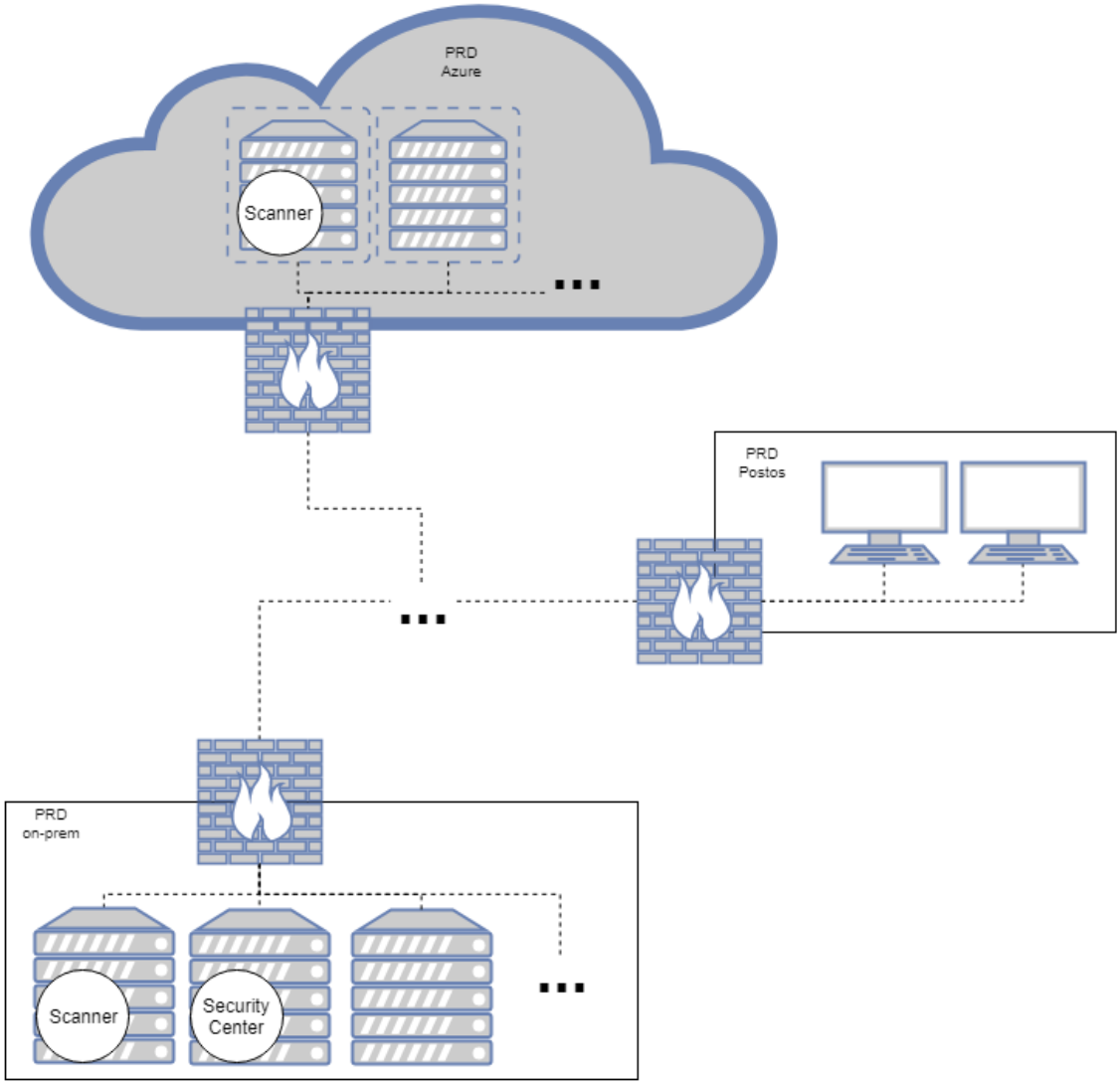

Figura 6 Arquitetura final do sistema de análise e gestão de vulnerabilidades

Na figura 6 vemos um diagrama simplificado da infraestrutura crucial do sistema nesta prova de conceito.

É possível compreender o fluxo geral que despoleta as ações de análise à infraestrutura bem como a monitorização do sistema no dia-a-dia. Dois *scanners*, um apoiado num servidor na *cloud* e um segundo *on-prem*, permitem uma visão de ambas as redes sem a necessidade de reduzir a segurança em cada rede. O *scanner* posicionado em Azure tem acesso à infraestrutura *cloud*, simplificada neste diagrama, sendo o único ponto para o qual é necessário permitir a comunicação em todos os portos analisados, isto é, não existe necessidade de permitir que máquinas presentes noutros segmentos de rede possam comunicar em todos os portos até aos segmentos em Azure. A única ligação que precisa de ser criada é entre o *scanner* e o Security Center, comunicação esta que é cifrada. O mesmo se verifica no caso *on-prem*. O *scanner* é a única máquina com visão no segmento de rede definido. Neste caso, encontra-se no mesmo segmento de rede da plataforma

Security Center, mas esta poderia ser criada em qualquer posição na rede e comunicaria da mesma forma com todos os *scanners*, de forma segura.

Considerando os requisitos de sistema previamente discutidos recomendados pela Tenable, foram determinadas as especificações necessárias para a realidade da organização. Não é expectável que o sistema tenha de suportar uma infraestrutura com a dimensão de 2500 *hosts* uma vez que não será utilizada para gerir todos os postos de trabalho que têm *desktops* mas apenas servidores de suporte aplicacional e de produtos. Assim, o número de dispositivos estará mais próximo de algumas centenas e não chegará aos milhares. A Tenable dispõe de *appliances* virtuais como imagens de sistema que permitem uma criação mais rápida dos *scanners* uma vez que as imagens já contêm os *scanners* Nessus pré-instalados bem como cumprem com os requisitos de *software* necessários ao funcionamento dos mesmos.

Tipologia	Infraestrutura	Núcleos CPU	Memória	Armazenamento	Sistema Operativo
Security Center	On-prem	4	8GB	256GB	CentOS Linux 7
Scanner	On-prem	4	8GB	96GB	CentOS Linux 7
Scanner	Azure	2	≈4GB*	≈275GB*	CentOS Linux 7

*Tabela 2 Informação sobre a infraestrutura aprovisionada para a prova de conceito (*Valores expressos em GiB pela Microsoft)*

Começando pela infraestrutura *on-prem*, a máquina que suporta o a plataforma central foi criada com base num servidor virtual CentOS Linux 7 com acesso aos requisitos mínimos recomendados para permitir que o número de *scanners* possa escalar e cobrir a dimensão de 2500 *hosts*, ainda que não seja necessário inicialmente. Esta máquina conta ainda com 256GB de capacidade de armazenamento.

O *scanner* montado *on-prem* é também uma máquina com sistema operativo CentOS, criado com a imagem providenciada pelo fabricante, que difere apenas da plataforma central na sua capacidade de armazenamento mais reduzida, uma vez que não guarda os registos das análises de vulnerabilidades por um longo período de tempo (essa tarefa cabe ao Security Center), nem faz avaliação de tendência, como a plataforma central faz.

Por último, o *scanner* disponibilizado em Azure foi criado com base nas opções existentes sendo escolhido uma máquina virtual com acesso a recursos limitados, uma vez que a infraestrutura em *cloud* é a menos expansiva. Esta máquina foi criada com base na série de VMs de utilização geral Av2, especificamente a verão A2_v2 que conta com 2 CPUs virtuais e 4 GiB de memória[15].

3.5.3 Posicionamento, comunicação e rede

O posicionamento escolhido para cada uma das máquinas foi determinado de forma a maximizar a visibilidade e alcance do sistema, pois ainda que se trate de uma versão para prova de conceito, tal como referido, esta infraestrutura tem como objetivo ser reaproveitada para o *deployment* de um sistema de análise e gestão de vulnerabilidades para toda a infraestrutura que serve as aplicações e soluções da organização.

A sonda em Azure foi colocada num segmento de rede dedicado. Embora possa parecer à partida uma escolha que limita o alcance deste *scanner*, a infraestrutura *cloud* da organização ainda se encontra numa fase algo embrionária. Cada solução existe na sua própria rede segregada e não existem redes de gestão comuns a todas as *subnets*. A solução perfeita para manter a completa segregação seria a criação de sondas em cada segmento de rede. No entanto, sendo esta prova de conceito apenas a base de um futuro sistema alargado, e considerando o investimento necessário à criação de máquinas virtuais em todos os segmentos, optou-se por permitir que a zona onde a sonda reside tenha comunicação com as restantes.

Quanto à sonda localizada *on-prem* e à plataforma central Security Center, colocar os dois de forma a maximizar o alcance do *scanner* torna-se uma tarefa mais difícil considerando o paradigma já utilizado na organização.

A infraestrutura nas instalações é muito mais antiga do que a existente na *cloud* e já sofreu inúmeras iterações ao longo dos anos. Estas alterações complicam a implementação das duas máquinas uma vez que a rede é complexa e conta com diferentes metodologias de divisão entre ambientes produtivos e não produtivos, e ainda entre as redes utilizadas no início do milénio e as mais recentes.

A última iteração da infraestrutura, sobre a qual as soluções mais recentes da organização estão montadas, permite facilmente o acesso às várias redes dentro desta, contando com zonas bem definidas de ambientes produtivos e não produtivos. Por outro lado, os serviços mais antigos que formam o *core* da estrutura bancária não seguem a mesma lógica e contêm apenas segregações lógicas entre máquinas de ambientes distintos, ou até máquinas fora de qualquer domínio.

Felizmente, e considerando a realidade da existência de diversos fornecedores de soluções que fazem a gestão de infraestruturas próprias, foi criada uma rede de gestão com acesso transversal às diferentes infraestruturas paradigmáticas, possibilitando o acesso a grande parte dos ativos. Esta rede não permite uma visão completa de tudo o que

existe *on-prem*, seja por limitações técnicas, mas também por questões regulamentares[8], mas torna-se a melhor solução para a criação das máquinas de *scan* e o Security Center.

Naturalmente, e não sendo utilizadas sondas para todas os segmentos segregados e respetivas soluções aplicacionais, existe um risco acrescido ao criar máquinas com acesso alargado a tantos sistemas (em último caso, uma análise necessita da abertura de comunicações em todos os portos entre as sondas e a infraestrutura testada). De forma a reduzir o risco deste nível de acesso na rede, as regras de *firewall* que o permitem, bem como as contas com permissões de administração das máquinas para efetuarem as análises necessárias, estarão sempre desativadas fora dos momentos de *scan*, e a sua gestão estará sujeita a controlos processuais (sobre a forma de uma norma, processo e procedimento de gestão de vulnerabilidades) e técnicos (recorrendo a sistemas de gestão de identidades privilegiadas existentes).

[8] Redes completamente segregadas como o caso da rede para comunicações SWIFT terão de ser abrangidas com o auxílio de novas sondas, com o alargamento da prova de conceito a um sistema final de análise e gestão de vulnerabilidades.

Capítulo 4

Avaliação do sistema

4.1 Cumprimento dos requisitos

A conclusão deste projeto deu-se com a implementação da prova de conceito do sistema de análise e gestão de vulnerabilidades e com a apresentação do funcionamento do mesmo ao cliente.

Considerando os requisitos explorados nos pontos de definição arquitetural e requisitos funcionais, exploramos agora o seu cumprimento pelo sistema implementado, e possíveis limitações que tenham surgido.

A arquitetura da solução instalada respeita a definição inicial, isto é, como representado na figura 6, as sondas foram colocadas em pontos estratégicos da rede, uma em Azure e outra numa rede de gestão *on-prem*, segregando as duas infraestruturas. Este posicionamento permitiu analisar os sistemas dos ambientes de qualidade de duas soluções aplicacionais, uma *on-cloud* e outra *on-prem* com dimensões semelhantes.

A escolha de análise dos ambientes de qualidade deve-se ao facto de tentar assemelhar o funcionamento do sistema o máximo possível a uma implementação final sem criar disrupção nos processos de negócio ao explorar um ambiente produtivo. Embora não seja possível garantir que o ambiente de qualidade é uma representação completamente fiel do ambiente de produção, no futuro, uma das vantagens do sistema de análise e gestão de vulnerabilidades será validar o "*gap"* entre os dois ambientes, garantindo que estes se encontram equiparados.

O acesso à plataforma de gestão Security Center foi integrado com o Active Directory da organização, permitindo a criação de vários níveis de acesso. Foram criados 3 níveis diferentes para a prova de conceito. Para a equipa de segurança foram dadas permissões de criação e acesso aos resultados de todos os *scans.* Às equipas de gestão das infraestruturas das duas soluções analisadas foi permitido o acesso à gestão de credenciais utilizadas pela plataforma para correr as análises (credenciais estas que nenhuma equipa tem visibilidade após serem introduzidas na plataforma). Por último, foi criado um nível de administração para gestão de utilizadores e acessos, e configurações de gestão do Security Center. Este último foi também gerido pela equipa de segurança na prova de conceito, mas na versão final o objetivo será integrar este nível de permissão num tipo de acesso *just in time* com recursos a ferramentas já existentes na organização que permitem

a atribuição temporária de níveis mais elevados de privilégios registando a sua utilização para posterior auditoria.

A utilização de credenciais para análise dos *hosts* foi testada de duas formas diferentes.

Na solução *on-prem*, em que todas as máquinas se encontravam no mesmo domínio, foi criada uma conta de serviço com privilégios de administração local, que se encontra ativa durante o *scan* e posteriormente é retirada. Esta solução é mais prática do ponto de vista de gestão da conta na plataforma Security Center, mas requer a implementação de métodos que assegurem a segurança desta conta única bem como a alteração recorrente da sua *password*.

No caso da infraestrutura analisada em Azure, nem todas as máquinas se encontravam no mesmo domínio, e neste caso foi utilizada a metodologia de criação extraordinária de contas para a análise, introduzidas na plataforma pela equipa que gere esta infraestrutura. Esta solução dificulta a gestão de contas privilegiadas para análises periódicas, mas, ao garantir que nenhuma informação é partilhada pela equipa de gestão de infraestrutura, poderá ser a solução mais segura para análises extraordinárias, fora da periodicidade normal.

Após a análise das infraestruturas, a plataforma regista a tipologia de cada um dos *hosts* e permite designar categorias personalizadas. A gestão dos repositórios de vulnerabilidades pode ser feita por solução, permitindo acessos diferenciados e segregados à informação disponibilizada pelo Security Center. Nas análises feitas no âmbito da prova de conceito, criaram-se categorias específicas para as duas soluções distintas que permitiram às equipas responsáveis pelas infraestruturas de cada uma ter acesso direto aos painéis tal como a equipa de segurança os vê. Isto significa que ao contrário do que seria feito previamente, em que a equipa de segurança analisaria os resultados e prepararia um relatório para entrega, as equipas podem antecipar imediatamente as intervenções que serão necessárias. A intervenção torna-se, portanto, mais fluída e menos sujeita a possíveis atrasos derivados de falhas de comunicação entre equipas.

A informação das vulnerabilidades é apresentada com níveis de criticidade associados. Estes níveis desdobram-se em duas vertentes: um nível de criticidade estático baseado no CVSSv2 conhecido (que pode ser classificado como Info, Baixo, Médio, Alto e Crítico)[16] e um nível dinâmico denominado Vulnerability Priority Rating. Este último organiza as vulnerabilidades conforme a probabilidade de existirem *exploits* para cada vulnerabilidade e a sua urgência de correção[17]. Estes dois tipos de classificação permitem às equipas saber que esforços de correção devem ser priorizados de uma forma transparente e objetiva. Anteriormente, ainda que fosse possível determinar os *patches* de

segurança em cada máquina, não era simples coordenar esforços de remediação uma vez que muitas vezes as equipas apenas tinham visibilidade de quanto tempo estariam atrasados ao nível de correções disponibilizadas pelos fabricantes, mas não tinham a noção exata do potencial de exploração corrigido em cada uma das atualizações.

Aliado ao último ponto, foi possível apoiar a intervenção das equipas de gestão dos sistemas ao longo da implementação de *patches* e criar análises subsequentes automatizadas para validar a correta implementação dos mesmos. Não foi possível, no entanto, validar o *gap* entre diferentes ambientes uma vez que, como referido, a prova de conceito apenas analisou ambientes de qualidade.

Embora a plataforma o permita, nas duas aplicações testadas não foi necessário gerir o risco de aceitação de vulnerabilidades por incompatibilidade ou impossibilidade de correção, sendo que não foi possível avaliar o comportamento do sistema neste caso excecional.

Após a aceitação e avaliação do cumprimento dos requisitos definidos, esta prova de conceito serviu de base à ampliação e implementação do sistema de análise e gestão de vulnerabilidades.

4.2 Deficiências encontradas

A prova de conceito demonstrou a capacidade do sistema de cumprir com os objetivos definidos inicialmente, ainda que não tenham sido testadas alguns dos casos de uso do sistema.

A grande limitação da prova de conceito deve-se ao impedimento, numa primeira fase, de analisar todos os ambientes que as aplicações utilizam, desde desenvolvimento a *disaster recovery* mas principalmente produção. Este tipo de análise representa mais desafios, uma vez que como foi referido, dependendo da idade da solução, a sua infraestrutura pode ter diferentes paradigmas quanto à utilização e localização de ambientes produtivos e não produtivos de forma segregada.

Adicionalmente, não foi possível avaliar os procedimentos que acompanham a utilização do sistema no seu todo. Uma vez que as análises realizadas não se trataram de análises periódicas, as ações necessárias que despoletam a abertura de regras de firewall, a criação ou ativação de contas de administração local das máquinas e o mapeamento dos recursos de cada solução, necessitaram sempre de intervenção manual, uma vez que os documentos normativos que estabelecem este funcionamento não se encontram finalizados, nem existe integração com o sistema de *tickets* da organização.

A não utilização de agentes, embora reduza o esforço na certificação de compatibilidade dos mesmos com as aplicações de cada máquina, pode tornar-se um obstáculo à medida que o âmbito da gestão de vulnerabilidades aumenta e cada vez mais análises periódicas são feitas. Uma vez que estes agentes resolveriam a necessidade de utilização de contas de administração local e reduziriam a necessidade de abertura de regras de *firewall*, é possível que a sua utilização seja uma evolução natural para o sistema.

Capítulo 5

Discussão e trabalho futuro

5.1 Discussão

O desenvolvimento do sistema exposto neste documento teve como objetivo a colmatação da necessidade de estruturar os esforços de gestão de vulnerabilidades na instituição bancária em que o mesmo foi construído e, adicionalmente, responder às dinâmicas necessidades regulamentares que se aplicam globalmente.

O sistema foi desenhado e implementado tendo em consideração estas particularidades e necessidades, evidenciando também o esforço que cada vez mais organizações têm de ter e a forma como a segurança informática ocupa uma posição de relevo nas suas preocupações.

O maior desafio à implementação deste sistema e a sua consequente avaliação envolve a inúmera quantidade de fatores que influenciam os processos de atualização e correção de vulnerabilidades. Existem muitos obstáculos a estes, como a idade das infraestruturas, das suas arquiteturas e o investimento necessário, seja em recursos humanos ou materiais, para garantir compatibilidade entre os sistemas existentes e reduzir ao máximo o impacto operacional. No entanto, a pressão regulamentar e o risco de exposição de informação sensível a atacantes leva a que cada vez mais se procure investir na proteção e prevenção contra ataques maliciosos informáticos.

Anteriormente a este projeto, também não existia na organização um processo formal de atualização e aplicação de *patches*, o que dificultou a introdução de um sistema de análise e gestão de vulnerabilidades que necessita que o mesmo tenha algum nível de maturidade.

É esperado que com a introdução do sistema desenvolvido neste projeto, primeiro em forma de piloto, e posteriormente abrangendo a maioria das soluções aplicacionais, o esforço necessário para correção de vulnerabilidades será cada vez menor e os casos de sistemas vulneráveis serão reduzidos drasticamente.

Certamente as decisões tomadas ao longo do desenvolvimento deste piloto não serão finais e o sistema evoluirá de forma dinâmica, para acomodar novos desafios técnicos e o regulamentares que possam surgir.

5.2 Trabalho futuro

Ao longo da implementação do piloto foram identificados possíveis incrementos e melhorias ao sistema de análise e gestão de vulnerabilidades.

Como foi referido, para poder funcionar de forma eficiente, a utilização deste sistema requer enquadramento em controlos técnicos e processuais. Isto significa que o trabalho feito na norma e no processo de gestão de vulnerabilidades e *patch management* deve continuar, e a estes deverá juntar-se um procedimento que determine a metodologia sistemática para realização de análises periódicas à infraestrutura, mas também análises realizadas em sede de projeto.

Adicionalmente, os controlos técnicos necessários para auxiliar a gestão de contas privilegiadas tanto para acesso à plataforma como para realização de análises nas máquinas, terão de ser implementados com recurso a ferramentas que estão a ser implementadas na organização em simultâneo, como o Azure AD Privileged Identity Management e o Microsoft Local Administrator Password Solution.

Seria uma mais-valia integrar as várias plataformas existentes na organização com este sistema, sendo que a plataforma utilizada para criação de *tickets* na organização, que permite solicitar abertura de regras de *firewall*, criação de contas e acessos, planear intervenções e correções na infraestrutura, pode comunicar com o Security Center e automaticamente tratar das solicitações necessárias.

O âmbito do sistema encontra-se de momento a ser ampliado de modo a abranger mais soluções da organização, em zonas de infraestrutura diferentes e com características particulares. Inicialmente optou-se por não utilizar uma solução com agentes na prova de conceito, mas, como referido, estes agentes podem reduzir o esforço de gestão de credenciais e comunicação na rede, pelo que a sua utilização futura deverá ser considerada.

No futuro, como referido, está previsto integrar o sistema de análise e gestão de vulnerabilidades com a ferramenta de SIEM da organização, uma vez que o Tenable Security Center permite, com sondas, motores de correlação de eventos e agentes, ter uma visão sobre as atividades existentes na rede.

Referências

[1] "OWASP Top 10-2017," 2003.

[2] R. Kissel, "Glossary of key information security terms," Gaithersburg, MD, May 2013.

[3] "Nessus | Tenable®." [Online]. Available: https://pt-br.tenable.com/products/nessus?tns_redirect=true. [Accessed: 27-Jul-2019].

[4] "Nmap: the Network Mapper - Free Security Scanner." [Online]. Available: https://nmap.org/. [Accessed: 27-Jul-2019].

[5] "Burp Suite Scanner - PortSwigger." [Online]. Available: https://portswigger.net/burp. [Accessed: 27-Jul-2019].

[6] "OpenVAS - OpenVAS - Open Vulnerability Assessment Scanner." [Online]. Available: http://www.openvas.org/. [Accessed: 27-Jul-2019].

[7] "Top Rated Vulnerability Management Software | Rapid7." [Online]. Available: https://www.rapid7.com/products/nexpose/. [Accessed: 27-Jul-2019].

[8] "Metasploit | Penetration Testing Software, Pen Testing Security | Metasploit." [Online]. Available: https://www.metasploit.com/. [Accessed: 27-Jul-2019].

[9] "Customer Security Programme (CSP) | SWIFT." [Online]. Available: https://www.swift.com/myswift/customer-security-programme-csp/security-controls. [Accessed: 28-Jul-2019].

[10] "Payment services (PSD 2) - Directive (EU) 2015/2366 | European Commission." [Online]. Available: https://ec.europa.eu/info/law/payment-services-psd-2-directive-eu-2015-2366_en. [Accessed: 18-Aug-2019].

[11] "SIBS - PSD2." [Online]. Available: https://www.sibs.pt/wp-content/uploads/sites/5/2018/01/SIBS_PSD2_20180111_final.pdf. [Accessed: 18-Aug-2019].

[12] "What ports are required for Tenable products?" [Online]. Available: https://community.tenable.com/s/article/What-ports-are-required-for-Tenable-products. [Accessed: 02-Jul-2019].

[13] "Port Requirements (Tenable.sc)." [Online]. Available: https://docs.tenable.com/sccv/5_9/Content/PortRequirements.htm. [Accessed: 01-Jul-2019].

[14] "Tenable.sc (Formerly SecurityCenter) Hardware Requirements (General Requirements)." [Online]. Available: https://docs.tenable.com/generalrequirements/Content/SCHardwareRequirements.htm. [Accessed: 29-Jun-2019].

[15] "Azure Linux VM sizes - General purpose | Microsoft Docs." [Online]. Available: https://docs.microsoft.com/en-us/azure/virtual-machines/linux/sizes-general. [Accessed: 01-Jul-2019].

[16] "CVSS v2 Complete Documentation." [Online]. Available: https://www.first.org/cvss/v2/guide. [Accessed: 10-Aug-2019].

[17] "Severity vs. VPR (Tenable.sc)." [Online]. Available: https://docs.tenable.com/sccv/Content/RiskMetrics.htm#VPR. [Accessed: 10-Aug-2019].